(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
21 November 2002 (21.11.2002)

PCT

(10) International Publication Number
WO 02/093368 A1

(51) International Patent Classification[7]: G06F 9/445

(21) International Application Number: PCT/KR02/00878

(22) International Filing Date: 10 May 2002 (10.05.2002)

(25) Filing Language: Korean

(26) Publication Language: English

(30) Priority Data:
2001/25924 11 May 2001 (11.05.2001) KR

(71) Applicant and
(72) Inventor: KIM, Yong-Hee [KR/KR]; A-905, Technopark, Boondang, 150 Yatop-dong, Boondang-ku, Sungnam-shi, Kyungki-do 468-816 (KR).

(74) Agent: PARK, Hyun Cheol; #310, ShinSeong B Building, 1589-14, Seocho-dong, Seocho-ku, Seoul 137-070 (KR).

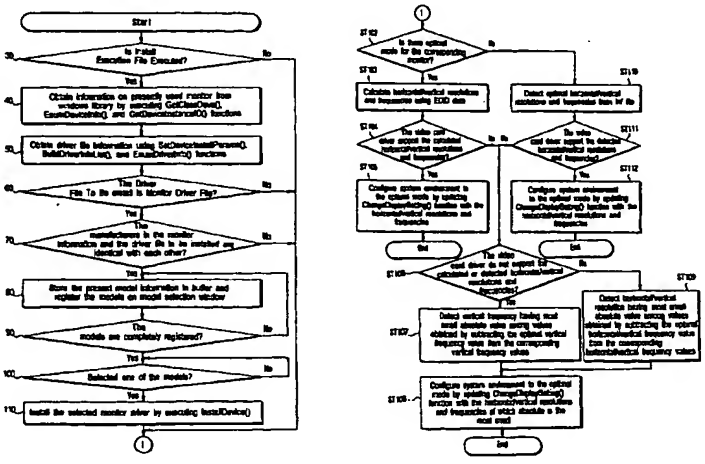
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: DRIVER AUTO-INSTALL METHOD FOR OPTIMIZING COMPUTER MONITOR SYSTEM AND STORAGE MEDIA HAVING PROGRAM SOURCE THEREOF



(57) Abstract: The present invention relates to a computer monitor system, and in particular, to a drive auto-install method for optimizing computer monitor system and storage media contained program source thereof. Once an installation execution file for automatically installing the monitor driver is, the CPU executes GetClassDevs(), EnumDeviceInfo(), GetDeviceInstanceID() functions, and etc. so as to collect information of a present-registered monitor and information of the driver file (Inf File) from an auxiliary storage device. After collecting the information, the CPU executes InstallDevice() function so as to install the selected monitor driver and optimally configures the computer system with the resolution, color, and refresh rate according to the characteristics of the monitor connected to the computer system after determining whether or not there exists optimal mode of the connected monitor and Inf file using the Extended Display Information Data (EDID) data.

WO 02/093368 A1

DRIVER AUTO-INSTALL METHOD FOR OPTIMIZING COMPUTER MONITOR SYSTEM AND STORAGE MEDIA HAVING PROGRAM SOURCE THEREOF

Technical Field

The present invention relates to a computer monitor system, and in particular, to a drive auto-install method for optimizing computer monitor system and storage media having program source thereof.

Background Art

Computer systems have become necessary item of life and most of the daily routines have been associated with the computer system in recent years. Generally, the computer system consists of a central processing unit and memory for accepting data and processing it into useful information, and input/output devices such as a monitor (screen), a keyboard, a mouse, etc.

However, the peripheral devices such as the monitor, keyboard, and etc. are manufacturer- and/or model-specific such that the computer system should be configured optimally for the specific devices when the devices are installed thereto. Especially, configuration for the monitor is significant because its mal-configuration makes a bad effect to human eyes, resulting in eyestrain and fatigue.

The monitor driver configuration is performed several configuration stages in regular order of selecting display, configuration, monitor, registration information, update, and confirmation in a control panel window. In case of manually configuring the monitor driver, an inf file containing configuration information is copied and stored in a register in order for the computer system

to be configured for the corresponding monitor model.

Since this monitor driver configuration process requires complicate configuration steps, it makes computer users feel complex.

In case of the manual configuration, the monitor driver may malfunction because the computer system is not automatically rebooted, and the rebooting manipulation may make the computer users feel troublesome.

Also, the operating systems (OS) of Windows 95, 98, and ME use 16-bit compatible library files and windows 2000 and XP uses 32-bit compatible library files such that an application program can not use two different versions of library files, resulting in utilization complication.

Disclosure of Invention

The present invention has been made in effort to solve the above problems and the object of the present invention is to provide a computer system optimization method capable of automatically installing a monitor driver and configuring color and refresh rate of the monitor according to characteristics of a corresponding monitor connected to the computer system with simple manipulation.

Another object of the present invention is to provide a storage media contained a program source implementing a method capable of automatically installing monitor driver for optimizing the monitor.

Still another object of the present invention is to provide a method for selectively installing a 16-bit and 32-bit library files with one execution file after determining kinds of windows system library files and storage media storing the method.

To achieve the above objects, the automatic monitor driver installation method according to one aspect of the present invention includes obtaining information of a presently used monitor from an installation file and an Extended Display Information Data (EDID) using a windows system library file when an installation execution command is inputted; calculating optimal values of a resolution and a vertical frequency on the basis of the obtained information and storing the optimal values in a region of a buffer; registering the stored information to a windows system register; and reconfiguring a computer system with an optimal mode of a corresponding monitor using the values of the resolution and vertical frequency.

According to another aspect of the present invention, the step of storing the information includes displaying only compatible monitors after checking vendor ID and product ID and also displaying only the driver files of which manufacturer is identical with that of monitor information obtained from the driver file (Inf File).

Also, the present invention provide a storage media contained program source for implementing the automatic monitor driver installation method.

Brief Description of Drawings

FIG. 1 is a block diagram showing a computer system according to a preferred embodiment of the present invention;

FIG. 2 is a flowchart illustrating a automatic driver installation method for optimizing a monitor according to a preferred embodiment of the present invention; and

FIG. 3 is an exemplary view illustrating a monitor driver file (Inf file).

Best Mode for Carrying Out the Invention

The preferred embodiment of the present invention will be described hereinafter with reference to the accompanying drawings. While describing the present invention, well known constructions and functions of the computer system that is thought make the key points of the present invention unclear will not be described.

FIG. 1 is a block diagram showing a computer system according to a preferred embodiment of the present invention.

As shown in FIG. 1, a computer system comprises a monitor 10 connected to a main body 20 with a cable and the monitor 10 is connected to a central processing unit (CPU) 23 of the main body 20 via an interface part 21 of a video card. Also, an auxiliary drive as an auxiliary storage is connected to the CPU 23 through an interface (not shown) thereof. This auxiliary drive can be a 3.5 inch floppy disc drive or a CD-ROM drive. A memory 24 contains program data, as shown in FIG. 2. That is, the CPU 23 controls the system such that a monitor driver can be automatically installed and optimized to the system on the basis of the monitor driver automatic install program data stored in the memory 24.

The automatic monitor driver-installing process for optimizing the computer system as structure above will be described hereinafter.

FIG. 2 is a flowchart illustrating an automatic driver installation method for optimizing a monitor according to a preferred embodiment of the present invention and FIG. 3 is an exemplary view illustrating a monitor driver file (Inf file).

In FIG. 2, once an install execution file (install.exe) for automatically installing the monitor driver is executed at step ST30, the CPU 23 executes GetClassDevs(), EnumDeviceInfo(), GetDeviceInstanceID() functions, and etc. so as to collect information of a present-registered monitor at step ST40. This information includes a model of the present monitor, which is important for changing the monitor driver. After collecting the information on the present-registered monitor, the CPU 23 executes SetDeviceInstallParams(), BuildDriverInfoList(), EnumDriverInfo() functions and etc. so as to collect driver file information (Inf file) from the auxiliary drive 22 and associated information from Extended Display Information Data (EDID) at step ST50.

Sequentially, the CPU 23 checks whether the driver file (Inf File) is a monitor driver file (Monitor Inf File) at step ST60. That is, the CPU 23 analyses the driver file and determines whether or not a device class is monitor class. If it is determined that the device class is for other device (such as HDD, Modem, and etc.), the CPU 23 ends the install method according to the preferred embodiment of the present invention. On the other hand, if it is determined that the driver file is the monitor driver file, the CPU 23 determines whether or not the manufacturer of monitor information collected at step 40 is identical with that of the monitor driver file (Monitor Inf File) to be installed, at step ST70. An exemplary monitor driver file used in the checking step is depicted in FIG. 3.

If it is determined that the monitor driver manufacturers of the presently used monitor driver and the monitor driver to be installed are identical to each other, the CPU 23 stores the model information in a buffer and displays a list of driver models in a driver model selection window in order for the user to select a

model to be installed at step ST80. If the computer system has a plug-and-play function, only the drivers compatible with the presently used monitor are displayed by comparing the vender IDs and manufacturer IDs. In this manner, since the compatible drivers among various monitor models are displayed, which make the user feel comfortable for selecting a model.

After the displaying the list of the driver models, the CPU 23 determines whether or not there is a model selection input from an input device (keyboard or mouse) at step ST100. If there is the model selection input, the CPU 23 executes an InstallDevice() function so as to install the selected monitor driver at step ST110.

Accordingly, the monitor driver installation can be comfortably performed with a 2-step process of monitor driver install instruction and model selection steps.

After installation complete, if an optimization program for reconfiguration of the computer system environment is executed by the user, the CPU 23 determines a value of second bit, which indicates whether or not a monitor optimization mode (Preferred Timing) exists, among a Feature Support Bit included in the DEID at step ST102.

The value "1" of the second bit means that the monitor supports the optimization mode and the value "0" means the monitor does not supports the optimization mode such that the CPU 23 calculates horizontal/vertical resolutions and horizontal/vertical frequencies of the corresponding monitor using Hactive, Hblanking, Vactive, Vblaking, and Pixel Clock of the EDID data if the value is 1 as follows.

Htotal = Hactive + Hblanking

Fh = PixelClock/Htotal

Vtotal = Vactive + Vblanking

Fv = Fh/Vtotal

After the horizontal/vertical resolutions and frequencies are calculated, the CPU 23 determines whether or not the video card, which is the interface part 21 responsible for signal processing between the monitor 10 and the main body 20, supports the calculated horizontal/vertical resolutions and frequencies at step ST104. by reading video card driver data from a register part 22, using GetRegData() APi function, and comparing the data with the calculated data.

If it is determined that the video card driver supports the optimal mode of the corresponding monitor, the CPU 23 updates variable of ChangeDisplaySetting() API function of the OS 24 with the horizontal/vertical resolutions and frequencies so as to reconfigure the computer system environment optimally to the corresponding monitor at step ST105.

On the other hand, if the driver does not support the optimal mode of the monitor, the CPU 23 reconfigures the computer system environment with mostly similar mode to the optimal mode among the modes supported by the driver.

That is, if the driver supports the horizontal/vertical resolutions but the vertical frequency, the CPU 23 reads all the vertical frequencies that the driver can support at the same resolution using the GetRegData() API function of the OS 24 and detects a vertical frequency having a most small absolute value among values obtained by subtracting the optimal vertical frequency value from the corresponding vertical frequency values at step ST107, and reconfigures the computer system environment near the optimal mode of the corresponding monitor by updating the variable of the ChangeDisplaySetting() API function

with the selected vertical frequency at step ST108.

On the other hand, in the driver does not support the horizontal/vertical resolutions, the CPU 23 reads all the horizontal/vertical frequencies that the driver can support using the GetRegData() API function of the OS 24 and detects a vertical frequency having a most small absolute value among values obtained by subtracting the optimal horizontal/vertical frequencies values from the corresponding horizontal/vertical frequencies values at step ST109, and reconfigures the computer system environment near the optimal mode of the corresponding monitor by updating the variable of the ChangeDisplaySetting() API function with the selected horizontal/vertical frequencies at step ST108.

If the CPU 23 can not read the EDID data stored in the register part 22 or there is no data on the optimal mode of the corresponding monitor, the CPU 23 calculates the optimal mode of the corresponding monitor using the monitor driver file (Inf File). The Inf File typically contains the optimal mode of the corresponding monitor, as shown in FIG. 3.

Accordingly, the CPU 23 can detect the optimal mode data of the corresponding monitor, i.e., horizontal frequency of 800, vertical frequency of 600, vertical frequency of 85, and etc, using the data of [800x600_85_16_100A] or HKR,,PreferredMode,,"800,600,85 contained in the Inf File at step ST110. If the video card driver support the optimal mode of the corresponding monitor calculated in this manner, the CPU 23 reconfigures the computer system environment optimal to the corresponding monitor by updating variables of the ChangeDisplaySetting() API function of the OS 24 with the detected horizontal/vertical resolutions and frequencies at steps ST111 and ST112.

On the other hand, if the driver supports the horizontal/vertical

resolutions but not vertical frequency, the CPU 23 reads all the vertical frequencies that the driver can support at the same resolution using the GetRegData() API function of the OS 24, detects a vertical frequency of which absolute value is the most small among the detected vertical frequencies at step ST107, and then reconfigures the computer system environment near the optimal mode of the corresponding monitor by updating variables of the ChangeDisplaySetting() API function of the OS 24 with the detected the vertical frequency at step ST108.

If the driver does not support the horizontal/vertical resolutions, the CPU 23 reads all the horizontal/vertical resolutions that the driver can support using the GetRdgData() API function of the OS 24, detects a vertical frequency which is the most small among the absolute values obtained by subtracting the optimal horizontal/vertical resolutions from the corresponding horizontal/vertical resolutions at step ST109, and then reconfigures the computer system environment near to the optimal mode of the corresponding monitor by updating the variables of the ChangeDisplaySetting() API function of the OS 24 with the obtained horizontal/vertical resolutions at step ST108.

As described above, in the present invention the monitor driver is automatically installed in a simple way different from the typical monitor driver installation method, which has complex installation steps, so as to enable the user to comfortably install the monitor driver. Also, since the resolution, color, and refresh rate of the monitor are optimally configured, in the present invention, according to the characteristics of the monitor connected to the main body of the computer by referring to the EDID data stored in the computer system register and the content of the Inf file, which is monitor driver file, it is possible to

prevent the monitor from being abnormally configured.

While this invention has been described in connection with what is presently considered to be the most practical and preferred embodiment, it is to be understood that the invention is not limited to the disclosed embodiments, but, on the contrary, is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims.

Claims

1. An automatic monitor driver installation method comprising the steps of:

obtaining information on a presently used monitor using a windows system library file when an installation execution command is inputted;

storing the obtained information in an allocated region of a buffer according to content of the information;

registering the stored information to a windows system register.

2. The automatic monitor driver installation method of claim 1 further comprises the step of selecting one of 16-bit and 32-bit library files.

3. The automatic monitor driver installation method of claim 1 wherein the step of storing the information includes displaying only compatible monitors after checking vendor ID and product ID.

4. The automatic monitor driver installation method of claim 1 wherein only driver files of which manufacturers are identical with that of the monitor information are displayed on a model selection window.

5. An automatic driver installation method for monitor optimization comprising the steps of:

obtaining information of a presently used monitor from an installation file and an Extended Display Information Data (EDID) using a windows system

library file when an installation execution command is inputted;

calculating optimal values of a resolution and a vertical frequency on the basis of the obtained information and storing the optimal values in a region of a buffer;

registering the stored information to a windows system register; and

reconfiguring a computer system with an optimal mode of a corresponding monitor using the values of the resolution and vertical frequency.

6. The method of claim 5 further comprises the step of configuring the computer system with horizontal and vertical frequencies mostly near to the optimal mode when a video card does not support the optimal mode of the monitor after registering the information to the windows system.

7. A storage media contained program source for implementing the automatic monitor driver installation method of any of claims 1 to 6.

FIG.1

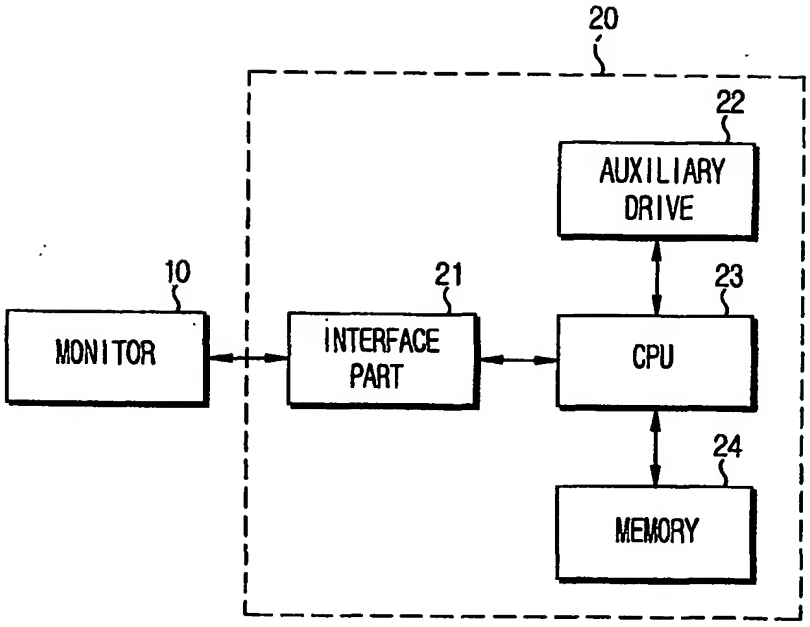

20
22
AUXILIARY DRIVE
10
21
23
MONITOR
INTERFACE PART
CPU
24
MEMORY

FIG.2A

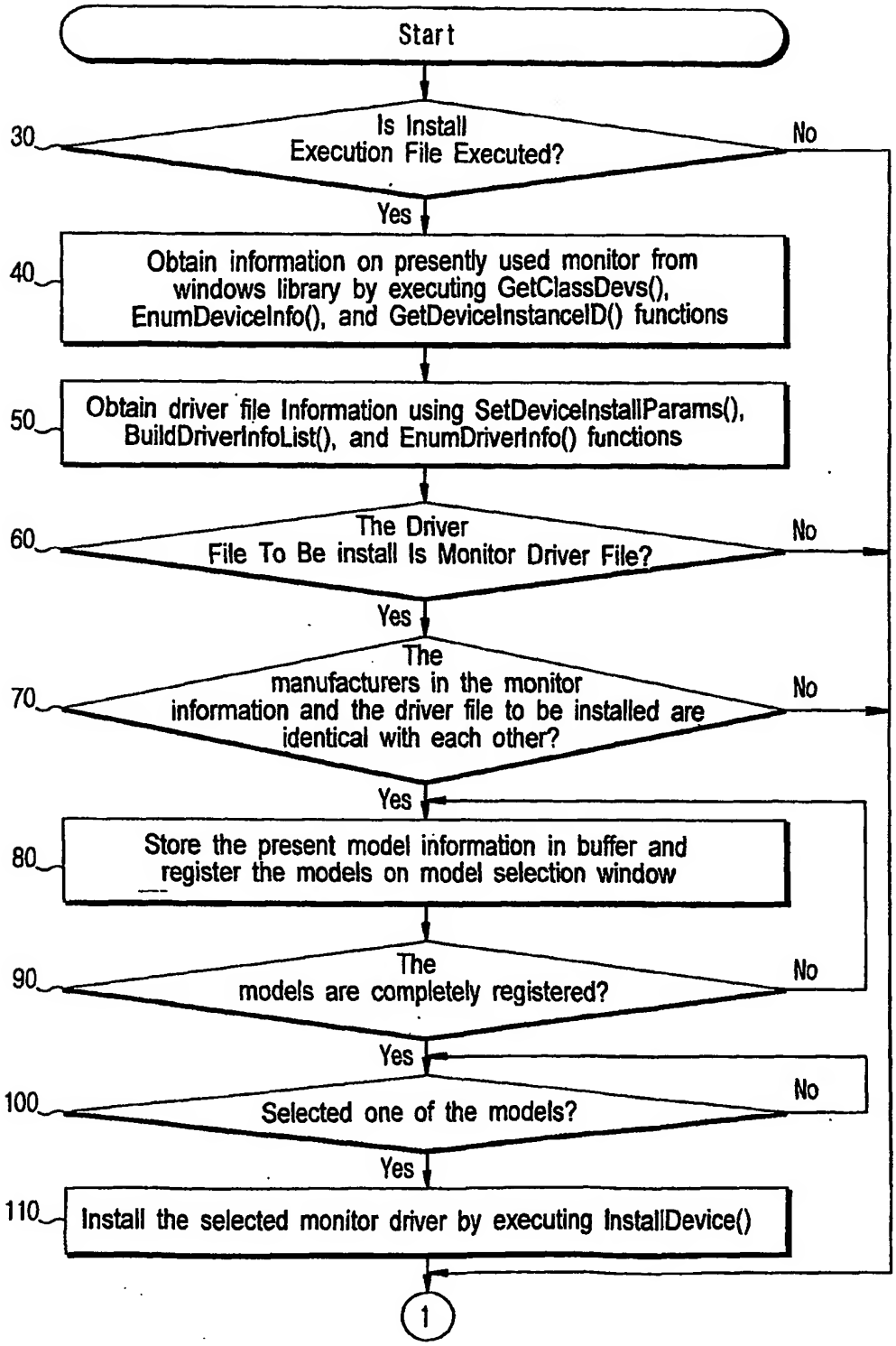

Start
30
Is Install Execution File Executed?
No
Yes
40
Obtain information on presently used monitor from windows library by executing GetClassDevs(), EnumDeviceInfo(), and GetDeviceInstanceID() functions
50
Obtain driver file Information using SetDeviceInstallParams(), BuildDriverInfoList(), and EnumDriverInfo() functions
60
The Driver File To Be install Is Monitor Driver File?
No
Yes
70
The manufacturers in the monitor information and the driver file to be installed are identical with each other?
No
Yes
80
Store the present model information in buffer and register the models on model selection window
90
The models are completely registered?
No
Yes
100
Selected one of the models?
No
Yes
110
Install the selected monitor driver by executing InstallDevice()
1

FIG.2B

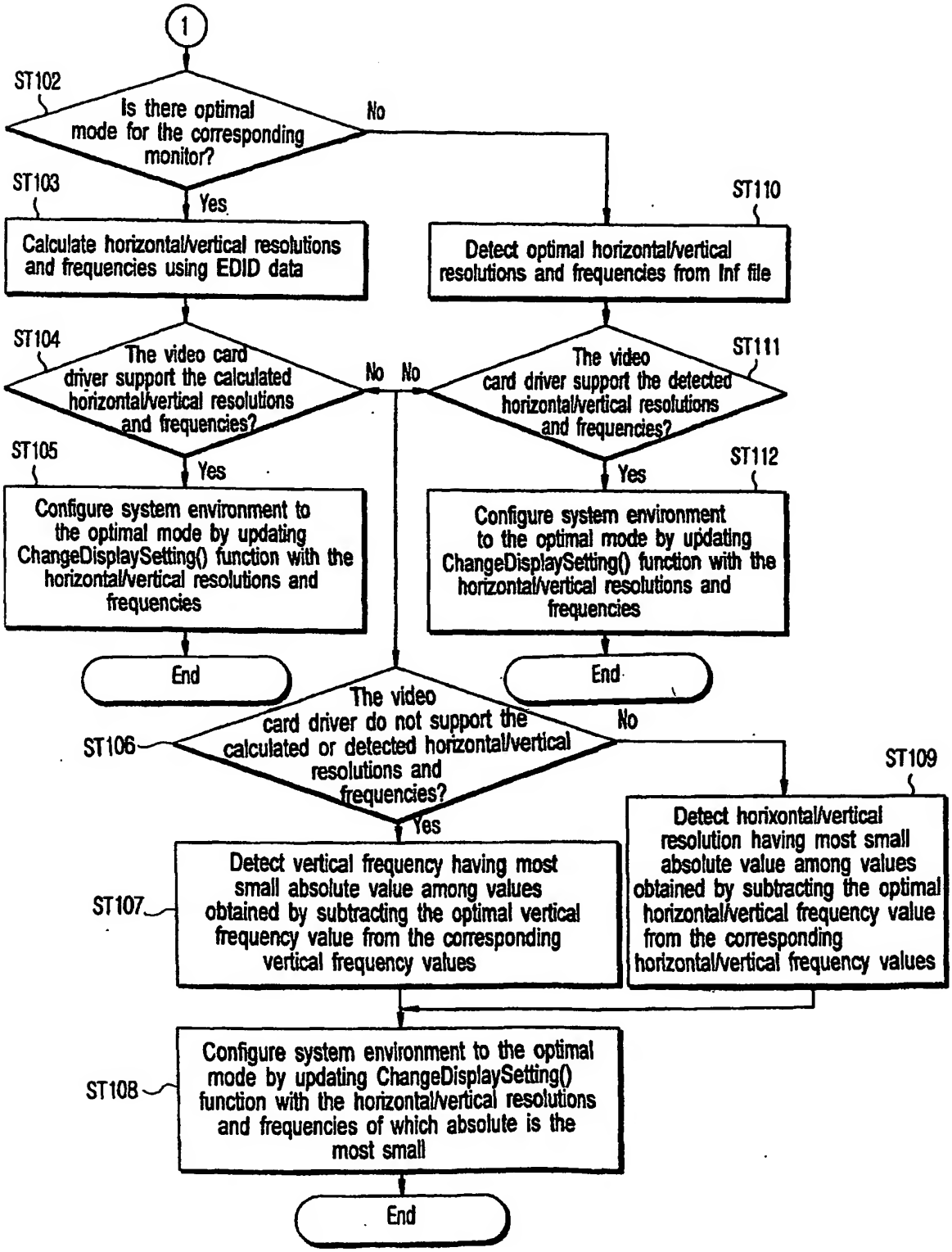

1
ST102
Is there optimal mode for the corresponding monitor?
No
Yes
ST103
Calculate horizontal/vertical resolutions and frequencies using EDID data
ST110
Detect optimal horizontal/vertical resolutions and frequencies from Inf file
ST104
The video card driver support the calculated horizontal/vertical resolutions and frequencies?
No
No
ST111
The video card driver support the detected horizontal/vertical resolutions and frequencies?
ST105
Yes
Yes
ST112
Configure system environment to the optimal mode by updating ChangeDisplaySetting() function with the horizontal/vertical resolutions and frequencies
Configure system environment to the optimal mode by updating ChangeDisplaySetting() function with the horizontal/vertical resolutions and frequencies
End
End
ST106
The video card driver do not support the calculated or detected horizontal/vertical resolutions and frequencies?
No
Yes
ST109
Detect horixontal/vertical resolution having most small absolute value among values obtained by subtracting the optimal horizontal/vertical frequency value from the corresponding horizontal/vertical frequency values
ST107
Detect vertical frequency having most small absolute value among values obtained by subtracting the optimal vertical frequency value from the corresponding vertical frequency values
ST108
Configure system environment to the optimal mode by updating ChangeDisplaySetting() function with the horizontal/vertical resolutions and frequencies of which absolute is the most small
End

FIG.3

```
[version]
signature="$CHICAGO$"
Class=Monitor
ClassGuid={4D36E96E-E325-11CE-BFC1-08002BE10318}
Provider=%MasTech%

[DestinationDirs]
DefaultDestDir=11
100A.CopyFiles=23
200B.CopyFiles=23

[SourceDisksNames]
1="MasTech Monitor Installation Disk",,1

[SourceDisksFiles]
CD100A.icm=1
CD200B.icm=1

[Manufacturer]
%MasTech%=MasTech

[Samsung]
%100A%    =800x600_85_16_100A, Monitor\MST1064
%200B%    =1024x768_60_16_200B, Monitor\MST1067

;
; Install Section
; !!!! The length of Install Section String is 31 character. !!!!
[800x600_85_16_100A]
DelReg=DEL_CURRENT_REG
AddReg=100A.AddReg, 800, DPMS
CopyFiles=100A.CopyFiles
[1024x768_60_16_200B]
DelReg=DEL_CURRENT_REG
AddReg=200B.AddReg, 1024, DPMS
CopyFiles=200B.CopyFiles
;-----------------------------------------------------
[DEL_CURRENT_REG]
HKR,MODES
HKR,,MaxResolution
HKR,,DPMS
HKR,,ICMProfile

[640]
HKR,,MaxResolution,,"640,480"
[800]
HKR,,MaxResolution,,"800,600"
[1024]
HKR,,MaxResolution,,"1024,768"
[1280]
HKR,,MaxResolution,,"1280,1024"
[1600]
HKR,,MaxResolution,,"1600,1200"

[DPMS]
HKR,,DPMS,,1
;
;-----------------------------------
; Add Registry Section
[100A.AddReg]
HKR,"MODES\1024,768",Mode1,,"30-55,50-120,-,-"
HKR,,PreferredMode,,"800,600,85"
HKR,,ICMProfile,0,"CDP14B.icm"
[200B.AddReg]
HKR,"MODES\1024,768",Mode1,,"30-61,50-120,-,-"
HKR,,PreferredMode,,"1024,768,60"
HKR,,ICMProfile,0,"CDP15S.icm"
-------------------------------------
[100A.CopyFiles]
100A.icm
[200B.CopyFiles]
200B.icm
;-------------------------------------
[Strings]
MonitorClassName="Monitor"
MasTech="MasTech"
100A=" MasTech  100A"
200B=" MasTech  200B"
```

INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR02/00878

A. CLASSIFICATION OF SUBJECT MATTER

IPC7 G06F 9/445

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC7 G06F 9/445

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,748,980 A(Microsoft Corpration) 5, May 1998 see the whole document	1-7
Y	US 5,353,432 A(Compaq Computer Corporation) 4, Oct 1994 see the whole document	1-7
P,Y	KR 2002-14364 A(Samsung Corporation) 25, Feb 2002 see the whole document	5-7
A	KR 2002-330532 B(Ineternational Business Machines Corporation) 8, Aug 2002 see the whole document	1-7

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevence
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevence; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevence; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
06 SEPTEMBER 2002 (06.09.2002)	06 SEPTEMBER 2002 (06.09.2002)

Name and mailing address of the ISA/KR	Authorized officer
Korean Intellectual Property Office 920 Dunsan-dong, Seo-gu, Daejeon 302-701, Republic of Korea Facsimile No. 82-42-472-7140	YOO, Byung Chul Telephone No. 82-42-481-5678

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/KR02/00878

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5,748,980	5. 05. 1998	US 5655148 A	5. 08. 1997
		US 5787246 A	28. 07. 1998
		US 5793979 A	11. 08. 1998
		US 5809329 A	15. 09. 1998
		US 5819107 A	6. 10. 1998
		US 6003097 A	14. 12. 1999
US 5,353,432	4. 10. 1994	BE 358304 R2	14. 03. 1990
		BE 358304 R3	23. 10. 1991
		CA 1322057 A1	07. 09. 1993
		CH 358304 R2	14. 03. 1990
		CH 358304 R3	23. 10. 1991
		DE 358304 R2	14. 03. 1990
		DE 358304 R3	23. 10. 1991
		EP 358304 A2	14. 03. 1990
		EP 358304 A3	23. 10. 1991
		ES 358304 R2	14. 03. 1990
		ES 358304 R3	23. 10. 1991
		FR 358304 R2	14. 03. 1990
		FR 358304 R3	23. 10. 1991
		GB 358304 R2	14. 03. 1990
		GB 358304 R3	23. 10. 1991
		GR 358304 R2	14. 03. 1990
		GR 358304 R3	23. 10. 1991
		IT 358304 R2	14. 03. 1990
		IT 358304 R3	23. 10. 1991
		JP 2114352 A2	26. 04. 1990
		LI 358304 R2	14. 03. 1990
		LI 358304 R3	23. 10. 1991
		NL 358304 R2	14. 03. 1990
		NL 358304 R3	23. 10. 1991
		SE 358304 R2	14. 03. 1990
		SE 358304 R3	23. 10. 1991
		US 5161102 A	03. 11. 1992
		US 5257387 A	26. 10. 1993
		US 5263148 A	16. 11. 1993
KR 2002-14364	25. 02. 2002	None	
KR 2002-330532	08. 08. 2002	JP 10260842 A2	29. 09. 1998